



Institute of information technology, Allahabad

Btech 4th semester 2023 - Department of Information Technology

Course: Software engineering

Group project – P2

Software Design Specification (SDS)

SERL LAB

By Team ELEMENTO:

IIT2021272: Aman Nanda

IIT2021234: Sindhu

IIB20221034: Saurav Kumar

IIB2021018: Ashwani Jha

IIT2021217: Satyam Thakur

Table of Contents

1. Introduction.....	4
1.1 Purpose.....	4
1.2 System Overview	5
1.3 Supporting Materials.....	5
1.4 Definitions and Acronyms.	5
2. Design Considerations.	6
2.1 Assumptions.....	6
2.2 Constraints.	7
2.3 System Environment.	8
3. Architecture.....	7
4. Database Schema.	8
5. User Interface Design.	9
6. Conclusion... ..	11

1. Introduction

This design will detail the implementation of the requirements as defined in the Software Requirements Specification

The Software Engineering Research Lab at IIITA (Indian Institute of Information Technology, Allahabad) is a dedicated facility focused on advancing the field of software engineering through cutting-edge research and development. This lab serves as a hub for collaboration, innovation, and knowledge dissemination among researchers, faculty members, students, and industry professionals

1.1 Purpose

The purpose of a software engineering research lab at IIITA (Indian Institute of Information Technology, Allahabad) can vary based on the specific objectives and focus areas of the lab. However, here are some common purposes for a software engineering research lab:

1. **Advancing Knowledge:** The lab aims to contribute to the field of software engineering by conducting research that advances theoretical understanding, identifies new methodologies, and develops innovative solutions to challenges in software development.
2. **Innovation and Technology Transfer:** The lab focuses on exploring and creating novel technologies, tools, and techniques in software engineering. It aims to transfer these innovations to industry and society, fostering technological advancements and practical applications.
3. **Collaborative Research:** The lab provides a platform for collaboration among researchers, both within IIITA and with external partners. It promotes interdisciplinary research by engaging experts from various domains, such as computer science, human-computer interaction, artificial intelligence, and software analytics.
4. **Industry Collaboration:** The lab fosters partnerships and collaborations with industry organizations, facilitating knowledge exchange, technology transfer, and joint research projects. This collaboration aims to address real-world software engineering challenges and bridge the gap between academia and industry.
5. **Graduate and Postgraduate Education:** The lab plays a crucial role in training and mentoring graduate and postgraduate students in software engineering research. It provides a hands-on learning environment where students can gain practical research experience, contribute to ongoing projects, and develop their research skills.
6. **Publication and Dissemination:** The lab aims to publish research findings in reputable conferences and journals, contributing to the body of knowledge in software engineering. It also promotes the dissemination of research outcomes through technical reports, workshops, and seminars, fostering intellectual discourse and knowledge sharing.

These purposes highlight the core objectives of a software engineering research lab at IIITA, which revolve around advancing knowledge, promoting innovation, collaborating with industry, and nurturing the next generation of software engineering researchers and practitioners.

1.2 System Overview:

The SERL lab will consist of the following key components:

1. User Interface (UI): The UI component will be responsible for providing an intuitive interface for users to interact with the application. It will enable users to set reminders, configure event details, manage their preferences, and view notifications.
2. Reminder Management: This component will handle the creation, modification, and deletion of reminders. It will allow users to set specific event details such as event name, time, and associated preferences (e.g., silent mode or ring mode).
3. Location Tracking: The location tracking component will utilize the device's GPS capabilities to determine the user's current location. It will continuously monitor the user's location and trigger notifications based on predefined event locations.
4. Notification System: The notification system will generate and deliver notifications to users based on their location and configured reminders. It will communicate with the UI component to display notifications to the user.
5. Profile Management: This component will handle user profiles and preferences. Users will be able to log in, set and change passwords etc.

1.3 Supporting materials:

Technology stack and frameworks to be used in the development of the NOTIFY© app

- Programming language: .
- Frame work : Node JS.
- Database : Sql.
- Android based application.

1.4 Definitions, Acronyms, and Abbreviations:

Acronyms and Abbreviations:

- 1.4.1 SERL: Software Engineering Research Lab
- 1.4.2 SRS: Software Requirement Specification
- 1.4.3 SDS: Software Design Specification

2. Design Considerations:

When designing a software and research lab website, there are several key design considerations to keep in mind. These considerations aim to create a user-friendly and informative website that effectively represents the lab's work and facilitates engagement with visitors. Here are some design considerations for a software and research lab website:

- **Clear Navigation:** Design an intuitive and easy-to-use navigation structure that allows visitors to quickly find the information they are looking for. Use clear and descriptive labels for navigation menus and organize the content logically.
- **Responsive Design:** Ensure the website is responsive and optimized for different devices and screen sizes. The website should adapt and provide a seamless user experience on desktop computers, laptops, tablets, and mobile devices.
- **Visual Appeal and Branding:** Create a visually appealing website that reflects the lab's branding and design guidelines. Use consistent colors, fonts, and imagery to reinforce the lab's identity and create a cohesive visual experience.
- **Engaging Homepage:** The homepage should provide a compelling introduction to the lab, highlighting its research areas, achievements, and unique selling points. Use visuals, concise text, and call-to-action elements to encourage visitors to explore further.
- **Content Organization:** Organize the website content in a structured and logical manner. Use headings, subheadings, and bullet points to break up text and improve readability. Group related content together and consider using categories or tags to facilitate browsing.
- **Showcase Research Projects:** Highlight the lab's research projects and their impact. Provide summaries, visuals, and links to in-depth project pages or publications. Showcase the lab's expertise and demonstrate its contributions to the field.
- **Team and Expertise:** Create a dedicated section to introduce the lab's team members, their roles, and areas of expertise. Include brief bios, photos, and links to individual profiles or publications.
- **Publications and Resources:** Provide access to the lab's publications, technical reports, white papers, or other relevant resources. Consider organizing them by category, date, or research area and provide easy download options or links to external repositories.
- **Contact Information and Inquiry Forms:** Make it easy for visitors to contact the lab by prominently displaying contact information, such as email addresses or phone numbers. Consider including inquiry forms or a dedicated contact page to facilitate communication.
- **News and Events:** Include a section for lab news, updates, and upcoming events. Highlight recent achievements, conference presentations, or industry collaborations. Provide a calendar or list of events the lab is hosting or participating in.
- **Multimedia Content:** Enhance the website's engagement by incorporating multimedia content. Use images, videos, infographics, or interactive elements to visually showcase the lab's work, demonstrations, or prototypes.
- **Accessibility:** Ensure the website adheres to accessibility standards to accommodate users with disabilities. Use appropriate color contrast, provide alternative text for images, and make the website navigable using keyboard-only controls.
- **Search Functionality:** Implement a search feature that allows visitors to search for specific content or keywords within the website. This can improve usability and help users find relevant information quickly.
- **Social Media Integration:** Incorporate links to the lab's social media profiles to encourage visitors to connect and follow the lab's updates and announcements.

By considering these design considerations, you can create a software and research lab website that

effectively communicates the lab's work, engages visitors, and provides a positive user experience.

Constraints:

When designing the SERL (Software Engineering and Research Lab) website, there are several constraints that need to be considered. Here are some possible constraints that could apply:

- **Branding and Design Guidelines:** The website design needs to align with the lab's branding and design guidelines, ensuring consistency with the lab's visual identity, colors, and logo usage.
- **User Experience and Accessibility:** The website should provide a user-friendly experience and comply with accessibility standards, ensuring that it can be accessed and used by people with disabilities.
- **Content Management System (CMS):** If the lab has a preferred CMS or existing infrastructure, the website design should be compatible with the chosen CMS, considering any limitations or requirements imposed by the system.
- **Responsiveness and Mobile Compatibility:** The website should be responsive and optimized for various devices, including desktops, laptops, tablets, and mobile phones, to provide a seamless experience across different screen sizes.
- **Technical Compatibility:** Constraints related to the lab's existing technical infrastructure, such as server configurations, hosting limitations, or programming languages, may impact the choice of technologies and development approaches for the website.
- **Integration with Existing Systems:** If the lab website needs to integrate with other systems or databases, constraints related to APIs, data formats, or authentication mechanisms need to be considered during the design process.
- **Security and Privacy:** The website should comply with security best practices and adhere to privacy regulations, especially when handling sensitive data or user information.
- **Performance and Scalability:** Constraints related to website performance and scalability, such as page load times, database optimization, and caching strategies, should be considered to ensure a smooth browsing experience, even with increased traffic or content growth.
- **Multilingual Support:** If the lab operates in a multilingual environment, constraints related to language support and localization may need to be considered in the website design and content management.
- **Maintenance and Updates:** Constraints related to website maintenance and updates, such as ease of content management, version control, and backup procedures, should be taken into account to ensure smooth ongoing operations.
- **Budget and Resources:** The available budget and resources allocated for website design and development can impose constraints on the scope, features, and complexity of the website.
- **Time Constraints:** Project timelines and deadlines may impact the website design and development, requiring prioritization and efficient implementation.

Considering these constraints will help ensure that the SERL lab website meets the necessary requirements, aligns with the lab's goals and constraints, and provides a positive user experience for visitors.

2.1 System Environment:

To support the operation of a software and research lab website, you would typically need the following system environment:

3. Web Hosting:

- **Web Server:** A web server software such as Apache HTTP Server, Nginx, or Microsoft Internet Information Services (IIS) to host the website and handle HTTP requests.
- **Domain Name:** A registered domain name for the website, allowing users to access it through a memorable web address.

4. Content Management System (CMS):

- **CMS Software:** A CMS platform such as WordPress, Drupal, or Joomla, which provides a user-friendly interface for managing website content, including pages, blog posts, images, and multimedia.

5. Programming Languages and Frameworks:

- **HTML, CSS, and JavaScript:** Core web development languages used to create the website's structure, layout, and interactivity.
- **Server-Side Programming:** Depending on the specific requirements, a server-side programming language like PHP, Python (with Django or Flask frameworks), Ruby (with Ruby on Rails), or Node.js may be necessary for dynamic website functionality and server-side processing.

6. Database:

- **Database Management System (DBMS):** If the website requires data storage and retrieval, you may need a DBMS such as MySQL, PostgreSQL, MongoDB, or SQLite to handle structured or unstructured data.

7. Security and Encryption:

- **SSL/TLS Certificate:** An SSL/TLS certificate to enable secure communication over HTTPS, encrypting data transmitted between the website and its users.
- **Web Application Firewall (WAF):** A WAF software or service to protect the website from common web attacks, such as SQL injection or cross-site scripting (XSS).

8. Responsive Design and Cross-Browser Compatibility:

- **Responsive Frameworks:** CSS frameworks like Bootstrap or Foundation that facilitate responsive web design, ensuring the website adapts and displays correctly across different screen sizes and devices.
- **Cross-Browser Testing:** Testing the website across various web browsers (e.g., Chrome, Firefox, Safari, Edge) to ensure consistent functionality and appearance.

9. Analytics and Monitoring:

- **Website Analytics:** Integrating web analytics tools like Google Analytics or Matomo to track website visitors, analyze traffic patterns, and measure website performance.
- **Error and Performance Monitoring:** Implementing tools such as New Relic, Sentry, or Datadog to monitor and receive alerts for errors, performance issues, or downtime.

10. Backup and Disaster Recovery:

- **Regular Backups:** Implementing automated backup processes to regularly create copies of the website's files and databases, ensuring data can be restored in case of data loss or system failure.

- Offsite Backup Storage: Storing backup copies in separate locations or cloud storage to mitigate the risk of data loss in the event of a physical disaster or server failure.

11. Version Control and Deployment:

- Version Control System: Utilizing a version control system (e.g., Git, Mercurial) to track and manage changes to the website's source code, facilitating collaboration and rollback options.
- Deployment Workflow: Establishing a workflow to manage website deployment, including staging environments, testing, and continuous integration/continuous deployment (CI/CD) practices for efficient updates and releases.

It's worth noting that the specific system environment may vary depending on the complexity and scale of the lab's website. Additionally, depending on the lab's requirements, additional tools or services may be needed for features like e-commerce, user authentication, or integration with external systems.

3. Architecture:

The architecture of a software and research lab website typically follows a client-server architecture, which is a common architectural pattern for web applications. In this architecture, the website is divided into two main components: the client-side and the server-side.

Client-Side: The client-side is responsible for rendering the user interface and handling user interactions within the web browser. It comprises the following elements:

- Front-end Technologies: HTML, CSS, and JavaScript are used to create the structure, styling, and interactivity of the website. Frameworks and libraries like React, Angular, or Vue.js may be utilized to enhance the development process.
- User Interface (UI): The UI components, including navigation menus, forms, multimedia content, and interactive elements, are created to provide an engaging and intuitive user experience.
- Client-Side Scripting: JavaScript is used to implement client-side logic, such as form validation, user input processing, and dynamic content rendering.
- Responsive Design: The website is designed to be responsive, ensuring it adapts and displays correctly across different devices and screen sizes.

Server-Side: The server-side handles the business logic, data processing, and interaction with databases or external systems. It consists of the following components:

- Web Server: A web server, such as Apache or Nginx, receives and handles incoming HTTP requests from clients, forwarding them to the appropriate server-side components.
- Application Layer: The application layer contains the server-side code responsible for processing requests, executing business logic, and generating dynamic content. It can be implemented using server-side programming languages like PHP, Python (with frameworks like Django or Flask), Ruby (with Ruby on Rails), or Node.js.
- Database Management System (DBMS): A DBMS, such as MySQL, PostgreSQL, or MongoDB, is used to store and retrieve structured or unstructured data relevant to the website, such as user information, research publications, or project details.
- APIs and Integrations: The server-side may include APIs or integrations with external systems, such as payment gateways, authentication services, or external data sources.
- Server-Side Scripting: Server-side scripting is used to process and validate user input,

interact with the database, generate dynamic content, and handle session management or user authentication.

Communication and Data Exchange: The client-side and server-side components communicate using standard protocols and formats, such as HTTP for request/response communication and JSON or XML for data exchange. AJAX (Asynchronous JavaScript and XML) or modern techniques like Fetch API or GraphQL may be employed to facilitate asynchronous communication and enhance user experience.

Additional Components:

- **Content Management System (CMS):** If the website utilizes a CMS like WordPress, Drupal, or Joomla, it provides a structured framework and administration interface for managing website content.
- **Third-Party Services and Libraries:** Various third-party services and libraries, such as analytics tools, social media integrations, or UI component libraries, may be utilized to enhance the website's functionality and performance.

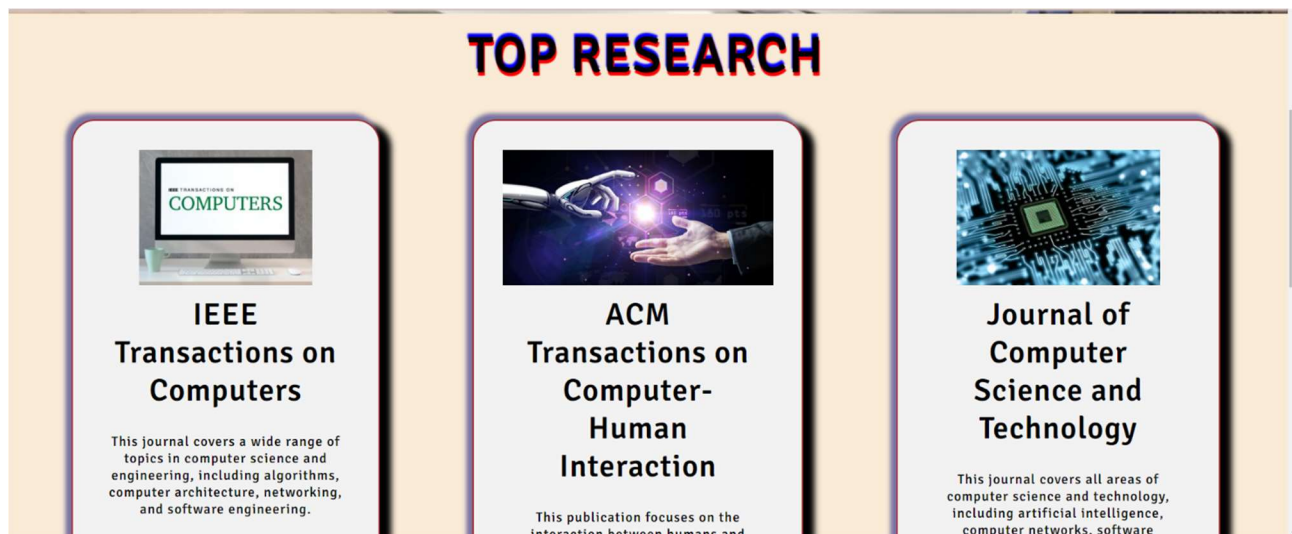
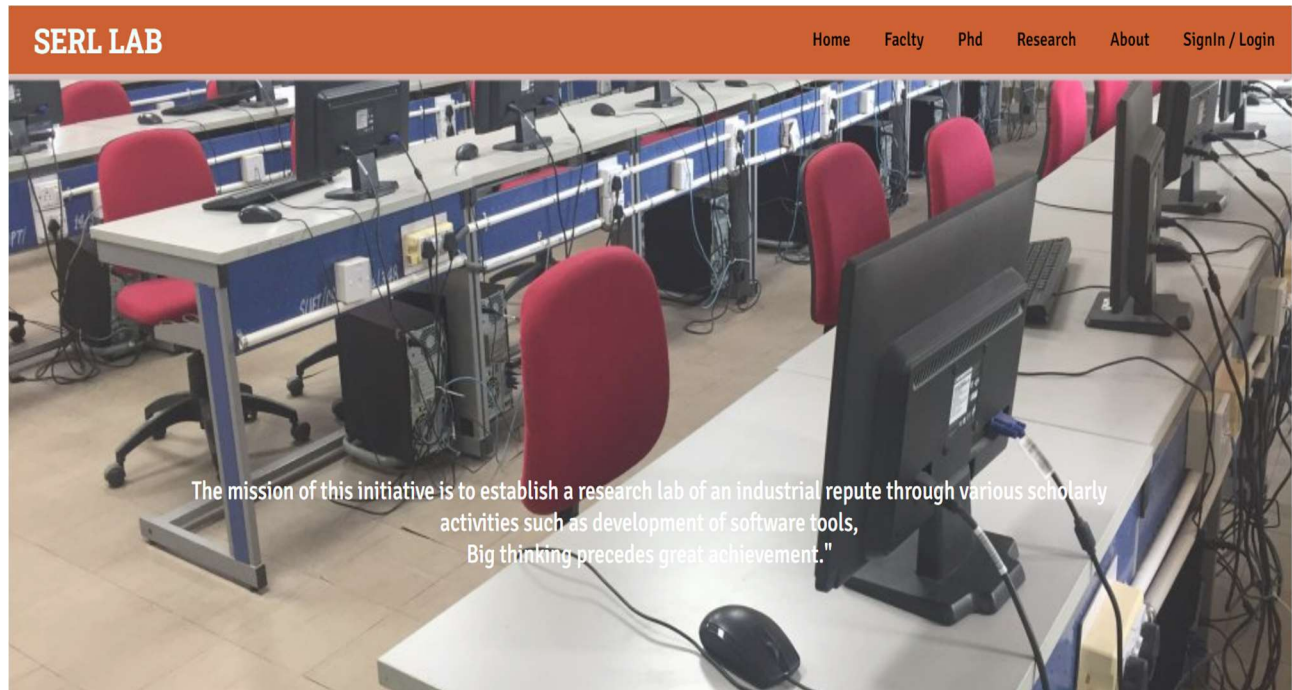
The architecture of a software and research lab website may vary depending on specific requirements and technologies chosen. However, the client-server architecture provides a flexible and scalable foundation for building web applications, allowing for the separation of concerns and facilitating modular development and maintenance.

4. DataBase Schema:

<div><div></div><div><pre>_id: ObjectId('6446612abdbde579af88733c') firstName: "Aman" lastName: "Nanda" email: "abc@gmail.com" password: "S2b\$10\$1XaqJLYYOheXgJuGGK6vkuL5Nam2VPRFQYYy3SweqI7W2Uivr54Ke" role: "Faculty" __v: 0</pre></div><div></div></div>	<div><div></div><div></div><div></div><div></div></div>
<div><div></div><div><pre>_id: ObjectId('644fe4b545fdb61510355081') firstName: "Saurav" lastName: "Kumar" email: "abd@gmail.com" password: "S2b\$10\$cA0Ajfyg6RWY8mzpVW2KPOCha6P8ltI3Oyivj/Xu2lATntCj.QTYS" role: "Phd" __v: 0</pre></div><div></div></div>	
<div><div></div><div><pre>_id: ObjectId('64421870cf2c283f0f74c783') description: "This is space project" link: "http://xya.com" mentor: "Sindhu" __v: 0</pre></div><div></div></div>	
<div><div></div><div><pre>_id: ObjectId('644ffa705a881a0fba6446d5') description: "This is much more intreasting project." link: "http://mlp.com" mentor: "xcbkj" publisher: "satyam" __v: 0</pre></div><div></div></div>	<div><div></div><div></div><div></div><div></div></div>
<div><div></div><div><pre>_id: ObjectId('6442177ccf2c283f0f74a28d') description: "This is a robotic project" link: "http://xyz.com" publisher: "Ashwani Jha" __v: 0</pre></div><div></div></div>	
<div><div></div><div><pre>_id: ObjectId('64466dea7a8c416ecde70abb') description: "hello" link: "http://qwe.com" publisher: "maa" __v: 0</pre></div><div></div></div>	
<div><div></div><div><pre>_id: ObjectId('6446754ef37d145e248f54dd') description: "djfkdsdhf" link: "http://ert.com" publisher: "maa" __v: 0</pre></div><div></div></div>	
<div><div></div><div><pre>_id: ObjectId('644fe45045fdb61510354369') username: "Anjali Gautam" email: "anjali@iiita.ac.in" address: "Flat No. 5 Ram nagarpur" position: "Assistent Profesor" > profileImage: Object > phone: 87867655687978 > website: Array > personalEmail: "gautam@gmail.com" > project: Array createdAt: 2023-05-01T16:09:52.715+00:00 updatedAt: 2023-05-01T16:09:52.715+00:00 __v: 0</pre></div><div></div></div>	

5. User Interface Design:

HOME PAGE



FOOTER

INVISIBLE DEVELOPER
SERL LAB
© copyright © All rights reserve 2023


CONTACT US

Contact us

SERL - Software Engineering Research Lab , CC-3
Indian Institute of Information Technology Allahabad
Devghat, Jhalwa, Allahabad-211015, U. P. INDIA
Phone: 91-532-2922000, Fax: 91-532-2922125,Email: contact@iiita.ac.in



FACULTY PAGE

Prof. Om Prakash Vyas
Professor Ph.D.
IIITA



Address:
Room No. 5003, Ground Floor Computer Center - III, Indian Institute of Information Technology, Allahabad Devghat, Jhalwa 211015, (UP) INDIA
IIITA
Contact No:
9153229225

Email:
opvyas@iiita.ac.in


Dr. Sonali Agarwal
Associate Professor Department of Information Technology Indian Institute of Information Technology Allahabad, Prayagraj, India
IIITA




Address:
Associate Professor Room No.: 5203, CC3 Building Indian Institute of Information Technology Allahabad Devghat, Jhalwa, Prayagraj(Allahabad) 211015, (UP) INDIA
IIITA
Contact No:
9132292242

Email:
raiagarwalsonali@gmail.com

PHD

Sakshi Singh IIITA	
	Address: IIITA Contact No: 91929394959
Email: rsi2015219@iiita.ac.in	

Rahul sharma IIITA	
	Address: IIITA Contact No: 9897969502
Email: phd2019198@iiita.ac.in	

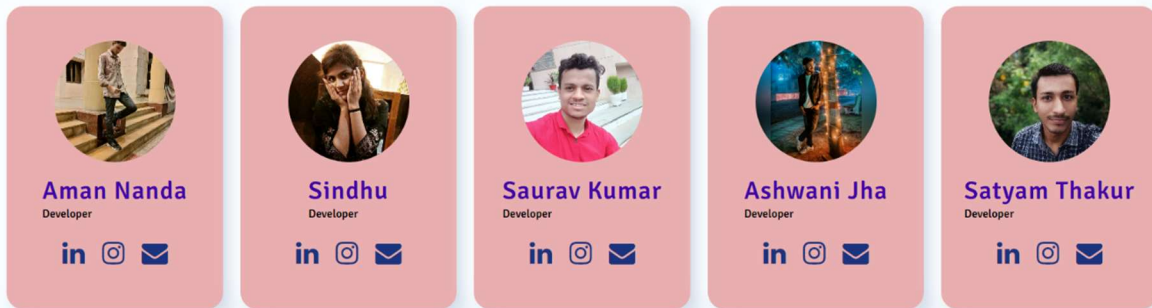
RESEARCH

[Faculty? share your project](#)

Description: The objective of this project is to build a computer controlled robot that can safely deliver an immobile person through an obstacle course in the shortest amount of time. To simulate real world situations, the robot must be able to climb a small ramp, cross a street without getting hit by a car, turn corners, fight off aggressive animals, climb stairs and free itself from a sandpit.	
Mentor: Mr. Abhishek Kumar	Publisher: Amit Kumar
View More	

Description: Machine learning is a sub-domain of computer science which evolved from the study of ' pattern recognition in data, and also from the computational learning theory in artificial intelligence. It is the first-class ticket to most interesting careers in data analytics today[1]. As data sources proliferate along with the computing power to process them, going straight to the data is one of the most straightforward ways to quickly gain insights and make predictions.	
Publisher: Anjali Gautam	
View More	

Meet the Team



6. Conclusion

This Software Design Specification document provides an overview of the SERL LAB outlining its components, architecture, and functional requirements. The document serves as a guide for the development of serl lab. It provides the overall design of our SERL LAB. It gives every detail of our design. It gives details about User Interface design, Database schema we used, and architecture we have built. A software design specification plays a crucial role in communicating, documenting, guiding development, verifying, and maintaining software systems. It enhances collaboration, enables scalability, and facilitates efficient development and future modifications.